

REMARKS**Overview of the Office Action**

Claim 6 was rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicant regards as the invention.

Claims 1-13, and 15-29 were rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107).

Claim 14 was rejected under 35 U.S.C. § 103(a) as being obvious over Chan in view of MSDN ("Lowering Total Cost of Ownership with Active Directory-Enabled Applications").

Status of the Claims/Amendments

Claims 1, 6, 15, 18, and 24 have been amended. Claims 1-29 are pending.

Claim Rejections Under 35 USC § 112**Regarding Claim 6:**

Claim 6 was rejected by the Examiner under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicant regards as the invention. Specifically, the Examiner cited the use of the phrase "assigning the interface for each interface" as being unclear. In response, Applicants have amended Claim 6 to replace the phrase "assigning the interface for each interface" with the phrase "assigning the interface identification for each interface". In light of this amendment, and in regard to Claim 6, Applicants respectfully request that this rejection under § 112 be withdrawn.

Claim Rejections Under 35 USC § 102(e)**Regarding Claims 1-13:**

Claims 1-13 were rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107). In response, Applicant has amended independent Claim 1 to include the additional element of aggregating the aggregatable software object to a directory services interface (item 212 in Fig. 2 of the present application).

In regard to independent Claim 1, Chan nowhere discloses aggregating the aggregatable software object to a directory services interface. As described in the present invention, a directory services interface is a distinct element that is different and separate from the directory services. Although Chan does disclose an “extending component” that enables “a client” to “control the defining of new object classes and new properties for directory services,” Chan nowhere describes any functionality pertaining to extending the “directory service system” itself by aggregating an aggregatable software object.

For the foregoing reason, Chan fails to teach or suggest all the claim elements necessary to anticipate the present invention under 35 U.S.C. § 102(e), and Applicant therefore respectfully requests that this rejection be withdrawn and that Claim 1 be allowed. Moreover, given that Claims 2-13 directly or indirectly depend from Claim 1, and given that claims which dependent upon an allowable claim are themselves inherently allowable, Applicants respectfully request that Claims 2-13 also be allowed.

Regarding Claims 15-17:

Claims 15-17 were also rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107). In response, Applicant has amended independent Claim 15 to

include the additional element of aggregating the aggregatable software object to a directory services interface (item 212 in Fig. 2 of the present application).

In regard to independent Claim 15, and as discussed previously herein, Chan nowhere discloses aggregating the aggregatable software object to a directory services interface. As described in the present invention, a directory services interface is a distinct element that is different and separate from the directory services. Although Chan does disclose an “extending component” that enables “a client” to “control the defining of new object classes and new properties for directory services,” Chan nowhere describes any functionality pertaining to extending the “directory service system” itself by aggregating an aggregatable software object.

For the foregoing reason, Chan fails to teach or suggest all the claim elements necessary to anticipate the present invention under 35 U.S.C. § 102(e), and Applicant therefore respectfully requests that this rejection be withdrawn and that Claim 15 be allowed. Moreover, given that Claims 16-17 directly depend from Claim 15, and given that claims which dependent upon an allowable claim are themselves inherently allowable, Applicants respectfully request that Claims 16-17 also be allowed.

Regarding Claims 18-23:

Claims 18-23 were also rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107). In response, Applicant has amended independent Claim 18 to include the additional element of aggregating the aggregatable software object to a directory services interface (item 212 in Fig. 2 of the present application).

In regard to independent Claim 18, and as discussed twice previously herein, Chan nowhere discloses aggregating the aggregatable software object to a directory services interface. As described in the present invention, a directory services interface is a distinct element that is

different and separate from the directory services. Although Chan does disclose an “extending component” that enables “a client” to “control the defining of new object classes and new properties for directory services,” Chan nowhere describes any functionality pertaining to extending the “directory service system” itself by aggregating an aggregatable software object.

For the foregoing reason, Chan fails to teach or suggest all the claim elements necessary to anticipate the present invention under 35 U.S.C. § 102(e), and Applicant therefore respectfully requests that this rejection be withdrawn and that Claim 18 be allowed. Moreover, given that Claims 19-23 directly or indirectly depend from Claim 18, and given that claims which dependent upon an allowable claim are themselves inherently allowable, Applicants respectfully request that Claims 19-23 also be allowed.

Regarding Claims 24-26:

Claims 24-26 were also rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107). In response, Applicant has amended independent Claim 24 to include the additional element of aggregating the aggregatable software object to a directory services interface (item 212 in Fig. 2 of the present application).

In regard to independent Claim 24, and as discussed thrice previously herein, Chan nowhere discloses aggregating the aggregatable software object to a directory services interface. As described in the present invention, a directory services interface is a distinct element that is different and separate from the directory services. Although Chan does disclose an “extending component” that enables “a client” to “control the defining of new object classes and new properties for directory services,” Chan nowhere describes any functionality pertaining to extending the “directory service system” itself by aggregating an aggregatable software object.

For the foregoing reason, Chan fails to teach or suggest all the claim elements necessary

to anticipate the present invention under 35 U.S.C. § 102(e), and Applicant therefore respectfully requests that this rejection be withdrawn and that Claim 24 be allowed. Moreover, given that Claims 25-26 directly depend from Claim 24, and given that claims which dependent upon an allowable claim are themselves inherently allowable, Applicants respectfully request that Claims 25-26 also be allowed.

Regarding Claims 27-29:

Claims 27-29 were also rejected under 35 U.S.C. § 102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,893,107). However, Applicant respectfully disagrees with the Examiner's conclusion that the invention of Chan includes each and every claim limitation present in these specific claims of the present invention.

In regard to independent Claim 27, Chan nowhere discloses a directory services interface extension. Although Chan does disclose—as the Examiner points out—functionality whereby “a client of the directory service system uses the extending component to define new object classes and properties,” it is important to note that, as described by Chan, “[t]he extending component controls the defining of new object classes and new properties for each **directory service**” (emphasis added). Chan, as well as the present invention, distinguishes a directory services interface (referred to as a “directory service system”) from the directory services, and therefore Chan merely discloses functionality for extending directory services but not the directory service system itself. In contrast, the “extension” referred to in the present claim is directed to the directory services interface and not the directory services, and thus the present claim is distinguishable from the invention of Chan.

For the foregoing reason, Chan fails to teach or suggest all the claim elements necessary to anticipate the present invention under 35 U.S.C. § 102(e), and Applicant therefore respectfully

requests that this rejection be withdrawn and that Claim 27 be allowed. Moreover, given that Claims 28-29 directly depend from Claim 27, and given that claims which dependent upon an allowable claim are themselves inherently allowable, Applicants respectfully request that Claims 28-29 also be allowed.

Claim Rejection Under 35 USC § 103(a)

Regarding Claim 14:

Claim 14 was rejected under 35 U.S.C. § 103(a) as being obvious over Chan in view of MSDN ("Lowering Total Cost of Ownership with Active Directory-Enabled Applications"). In response, Applicant has amended independent Claim 1, from which Claim 14 indirectly depends, to include the additional element of aggregating the aggregatable software object to a directory services interface (item 212 in Fig. 2 of the present application).

In order to establish a prima facie case of obviousness, three basic criteria must be met. First there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second there must be a reasonable expectation of success. Finally the prior art reference (or references when combined) must teach or suggest all the claim elements. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and cannot be based on applicant's disclosure. (MPEP §§ 2142, 2143.)

In regard to independent Claim 1, from which Claim 14 indirectly depends, neither Chan nor MSDN discloses aggregating the aggregatable software object to a directory services interface. As described in the present invention, a directory services interface is a distinct

element that is different and separate from the directory services. Although Chan does disclose an "extending component" that enables "a client" to "control the defining of new object classes and new properties for directory services," neither Chan nor MSDN describe any functionality pertaining to extending the "directory service system" itself by aggregating an aggregatable software object.

For the foregoing reason, the combination of Chan and MSDN fails to teach or suggest all the claim elements necessary to anticipate the present invention under 35 U.S.C. § 103(a), and Applicant therefore respectfully requests that this rejection be withdrawn and that Claim 14, which indirectly depends on Claim 1, be allowed.

CONCLUSION

Based on the reasons and rationale set forth herein, Applicants respectfully submit that the rejections have been overcome. Accordingly, Applicants request that the claims be allowed to issue. Should the Examiner have any questions, comments, or suggestions that would expedite the prosecution of the present case to allowance, Applicants' undersigned representative earnestly requests a telephone conference at (206) 332-1394.

Date: 3/17/03



Michael D. Stein
Registration No. 34,734
Attorney for the Applicant

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439

ATTACHMENT B
MARKED-UP VERSION OF AMENDED CLAIMS
(version with markings to show changes made)

Claims 1 and 6 have been amended as follows:

1. (amended) A computer-implemented method comprising:

A¹ inputting an aggregatable software object consistent with a predetermined software object framework and having a class identification and one or more interfaces, each interface having an interface identification; [and,]

 associating one of a directory class and a directory attribute to the class identification of the aggregatable software object, as stored in a predetermined location[.]; and

aggregating the aggregatable software object to a directory services interface.

6. (amended) The method of claim 1, wherein inputting an aggregatable software object comprises:

A² creating the aggregatable software object, including assigning the class identification to the aggregatable software object; and,

 creating and implementing the one or more interfaces of the aggregatable software object; including assigning the interface identification for each interface.

15. (amended) A computer-implemented method comprising:

A³ querying one of a directory class and a directory attribute to expose the one or more interfaces of the one of a directory class and a directory attribute, including one or more interfaces of an aggregatable software object having a class identification previously associated to the one of a directory class and a directory attribute;

 invoking one of the one or more interfaces of the aggregatable software object via the

A³ interface identification of the one of the one or more interfaces; [and,]

creating an instance of the aggregatable software object[.]; and

aggregating the aggregatable software object to a directory services interface.

18. (amended) A machine-readable medium having instructions stored thereon for execution by a processor to perform a method comprising:

A⁴ inputting an aggregatable software object consistent with a predetermined software object framework and having a class identification and one or more interfaces, each interface having an interface identification; [and,]

associating one of a directory class and a directory attribute to the class identification of the aggregatable software object, as stored in a predetermined location[.]; and

aggregating the aggregatable software object to a directory services interface.

24. (amended) A machine-readable medium having instructions stored thereon for execution by a processor to perform a method comprising:

querying one of a directory class and a directory attribute to expose the one or more

A⁵ interfaces of the one of a directory class and a directory attribute, including one or more

interfaces of an aggregatable software object having a class identification previously associated to the one of a directory class and a directory attribute;

invoking one of the one or more interfaces of the aggregatable software object via the interface identification of the one of the one or more interfaces; [and,]

creating an instance of the aggregatable software object[.]; and

aggregating the aggregatable software object to a directory services interface.
